**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC1/SC29/WG11**

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC1/SC29/WG11 MPEG2016/N16539**

**October 2016, Chengdu, CN**

|  |  |
| --- | --- |
| **Source** | **Requirements** |
| **Title** | **Draft requirements for network distributed video coding** |
| **Authors** | Jonatan Samuelsson, Xavier Ducloux |

# Introduction

This document contains draft requirements and application examples for network distributed video coding.

# Definition

In this document a network-distributed video coding system is defined as follows: A system for video encoding and decoding where processing is distributed across three or more processing units, wherein the processing units are interconnected through links with individual bandwidth constraints, and each processing unit has an individual processing capability. One of the processing units is the “original” encoder and one of the processing units is the “final” decoder.

# Scope

The scope of this document is to present requirements for video compression aspects of network-distributed video coding with focus on links and processing units that are traditionally not covered by video coding standards.

# Draft requirements

Specific requirements are proposed to efficiently support the application examples described in section 4. The requirements are expressed with reference to Figure 1, which illustrates a schematic view of one type of network distributed encoding system.



*Figure 1. Schematic view of network distributed encoding system.*

The draft requirements are as follows:

1. Stream T shall be able to convey information applicable to multiple different operating points and be significantly more effective compression-wise compared to conveying multiple independent compressed bitstreams for these operating points (simulcast). It shall be possible for the different operating points to differ in terms of image resolution, bitrate, frame rate, codec, profile/tier/level, color space, chroma subsampling format, and dynamic range.
2. The nominal operating point shall be conveyed as an independent compressed bitstream, while other operating points shall be conveyed as a dependent information from the nominal operating point.
3. The processing applied in System B in order to go from stream T to stream U shall be of significantly lower computational complexity than conventional transcoding (i.e. decoding followed by conventional re-encoding) without introducing degradation compared to conventional transcoding.
4. Additional delay introduced by the transcoding function shall be minimal so as to have negligible impact on overall system performance.
5. Stream U shall correspond to one of the multiple different operating points for which information have been conveyed in T and shall have minimal or no bitrate overhead compared to single layer coding for that operating point.
6. The processing applied in System C shall have minimal or no complexity increase over single layer decoding for each of the operating points.
7. The way information is conveyed shall use already standardized delivery formats and protocols.

The labels “S”, “V” and “System A” are included for completeness even though the proposed requirements do not reference them. “S” is the original video in raw or compressed format, “V” is the decoded video output from the decoder and “System A” is the original encoder capable of creating stream “T”. No specific requirements are envisioned for “S”, “V” and “System A” compared to conventional video compression schemes.

# Timeline

A preferred timeline is to finalize specification(s) for supporting network distributed video coding no later than finalization of next generation video codec. It should be considered to address specification(s) targeting existing video coding standards such as HEVC and AVC with a shorter timeline.

# Application examples

The following section contains three example applications that may benefit from network-distributed encoding. They all involve delivery of ABR (adaptive bitrate) video, where video content is provided in several different representations. Here, the representations may differ in terms of image resolution, frame rate, bitrate, as well as codec profile/level. They may also differ in terms of colour space, chroma subsampling format, dynamic range, codec and streaming formats.

## Storage of ABR video for on-demand delivery



*Figure 1. Application example: Storage of ABR video for on-demand delivery.*

Figure 1 shows an example of a system for preparation and delivery of ABR video. Original video (S) is ingested into the VoD or n-PVR preparation system. Purpose of the VoD or n-PVR preparation system is to prepare the video in a way that any specific ABR video representation can be delivered to the end user upon the user’s request, so that the end user’s equipment can decode (C) and render the video. In order to do so, the VoD or n-PVR preparation system applies an initial encoding, transforming the video into an initial stream (T), which is stored. At the time when a user requests a specific ABR representation, relevant parts of the stored stream (T) are retrieved and processed (B), generating the requested ABR video representation (U), which is then delivered to the end user’s equipment.

In order to maximize the end user experience, the finally delivered ABR video representation (U) needs to be compressed very efficiently, i.e. with minimal or no compression performance loss compared to direct encoding. At the same time, from an economical perspective, the cost for storage and processing needs to be minimized. One commonly employed solution today (1) performs initial encoding (A) and storage of independent ABR video representations. While that solution has very low processing (B) requirements, it can be very cost-intensive in terms of storage. Due to that, an alternative approach called just-in-time transcoding (2) has emerged, in which the initial encoding (A) provides only a single high quality ABR representation, and lower quality ABR representations are generated by the processing step (B) on-the-fly as the content is requested. While that solution is efficient in terms of storage, it poses very high processing (B) requirements. Also the transcoding operation introduces compression performance loss and thus degraded user experience.

Goals of network-distributed encoding in this application can be:

* to preserve ABR video representation quality as in independent ABR video representations storage solution (1) with a storage gain for an increase of processing (B),
* or to keep the advantage of storage efficiency (slight increase of storage) of the just-in-time transcoding solution (2) with a much lower processing (B) and no impact on the perceived quality.

## Delivery of live ABR video to multiple receivers in the home



*Figure 2. Application example: Delivery of live ABR video to multiple receivers in the home.*

Figure 2 shows an example of a system that delivers live ABR video to a user’s home. Within the home, several different ABR video representations are consumed by different devices. Original video (S) is ingested into the system and initially encoded (A) into a compressed stream (T), which is then delivered to the user’s home. Within the home, e.g. in the home gateway, processing (B) is applied to produce specific ABR video representations (U1, U2, U3) as requested by the different user devices. Note that the different ABR video representations (U1, U2, U3) may need to meet different device capabilities in terms of for example video resolution, codec support, profile/level support, and/or dynamic range support.

There are several potential bottlenecks within such system. Firstly, the delivery to the home is bandwidth limited (may be broadcast or multicast delivery). Therefore, in order to maximize the user experience, the stream that is delivered into the home (T) needs to be efficiently compressed. Secondly, the delivery within the home is bandwidth limited as well. Therefore, the streams (U) delivered within the home need to be efficiently compressed. Similar as in the example in Section 4.1, (1) simulcast delivery to the home and (2) single stream delivery to the home with transcoding in the home gateway are two potential solutions, having drawbacks in terms of compression efficiency for the delivery into the home (1), and in terms of processing cost and compression efficiency for delivery within the home (2), respectively.

In this application, network-distributed encoding tries to limit the bitrate overhead of the stream (T) delivered to the home compared to conventional transcoding (2) while operating with significantly lower processing in the home gateway (B) and providing equivalent or improved compression efficiency for delivery within the home.

## Network delivery of ABR video

### Delivery of live or on-demand ABR video inside a media delivery network



*Figure 3. Application example: Delivery of live or on-demand ABR video inside a media delivery network.*

Figure 3 shows an example of a media delivery network in which video content is prepared for live or on-demand ABR video delivery. Original video (S) is ingested and initially encoded into an initial stream (T) at an origin (A). The initial stream (T) is delivered to a node (B) within the media delivery network, up to the edge server, at which processing is applied in order to transform the initial stream (T) into ABR video representations (U1, U2, U3) that can be delivered to the end user devices (C). Note that the different ABR video representations (U1, U2, U3) may need to meet different device capabilities in terms of e.g.resolution, codec support, profile/level support, and/or dynamic range support.

While simulcast of ABR representations is an obvious and commonly used solution when preparing the initial stream (T) in this application example, there may be cases in which the bandwidth within the media delivery network is constrained and therefore more efficient compression of the initial stream (T) is desirable. A possible solution is to deliver only a high quality video representation within the media delivery network and generate the lower quality representations closer to the media network edge through transcoding (B). However, that solution may be economically inefficient due to the high processing cost associated with the transcoding. Moreover, the transcoding introduces compression performance loss, which affects the last-mile delivery to the end user and thus leads to degraded user experience.

In this application, network-distributed encoding can offer a much lower processing cost than conventional transcoding for a comparable compression performance for the last-mile delivery.

### Delivery of live or on-demand ABR video from a production facility to a distribution facility



*Figure 1. Application example: Delivery of live or on-demand ABR video from a production facility to a distribution facility.*

Figure 1 shows an example of a media production and distribution system in which video content is prepared for live or on-demand ABR video delivery. Original video (S) is ingested at a production facility and initially encoded (A) into an initial stream (T). The initial stream (T) is uploaded over a contribution link to a possibly cloud-based distribution facility, at which processing is applied in order to transform the initial stream (T) into ABR video representations (U1, U2, U3) that can be delivered over a delivery network to the end user devices (C). Note that the different ABR video representations (U1, U2, U3) may need to meet different device capabilities in terms of e.g. codec support, profile/level support, and/or dynamic range support.

A first existing technical solution for this application example is to deliver a high quality video over the contribution link, and to generate the lower quality representations at the distribution facility through transcoding (B). In order to minimize transcoding-induced compression efficiency losses, the bit rate for coding the high quality video that is sent over the contribution link may need to be very high, which may be conflicting with technical or economical bandwidth constraints on the contribution link. A second existing technical solution for the application example is that the different ABR video representations (U1, U2, U3) are generated already in the initial encoding step, and uploaded as simulcast over the contribution link. However, similar as in the first technical solution, since the bandwidth of the contribution link may be constrained, more efficient compression of the initial stream (T) is desirable.